

REMARKS

The rejections presented in the Office Action dated May 6, 2004 have been considered.

Claim 3 is amended to correct the typographical error of the double wording of "that".

The Office Action does not establish that claims 1, 2, and 15 are anticipated under 35 U.S.C. 102(e) by US Patent No. 6,665,671 to Coutant et al. (hereinafter "Coutant"). The rejection is respectfully traversed because the Office Action fails to show that all the limitations of the claims are taught by Coutant.

Claim 1 sets forth a method of optimizing an executable program having a plurality of functions. At least one of the functions has a first name associated with executable code that implements the function at a first address, and at least one linkage stub code segment has code that branches to the first address and a symbolic name by which the function is invoked in the program. The method includes identifying branch instructions having target addresses that reference the linkage stub code segment; and replacing the target addresses of the branch instructions with the first address. The Office Action clearly fails to show that these limitations are taught by Coutant.

The Office Action is mistaken in citing various sections of Coutant as teaching the limitations of claim 1 because Coutant's system is directed to optimizing access to shared data, not optimizing program code with function calls as claimed. Coutant's Background and Summary make clear that Coutant's approach deals with optimizing shared data access, not optimizing function calls. For example, Coutant's Background describes both remote code access, including function calls (col. 1, ll. 33-48) and remote data access (col. 1, ll. 49-67). These are clearly two different concepts. Coutant's Summary goes on to explain that his invention is for providing a method for optimizing and efficiently accessing shared data (col. 2, ll. 14-36). Coutant describes what remote function calls are, but does not describe any approaches for optimizing these calls. As the cited sections of Coutant clearly indicate and explained below, Coutant's approach is for remote data access, not optimizing function calls.

The Office Action mistakenly alleges that Coutant's col. 3, ll. 18-20 teach the limitations of identifying branch instructions having target addresses that reference the linkage stub code segment. The alleged teaching is wrongly applied because this section of Coutant simply describes a remote function call. There is no teaching or suggestion that Coutant's process in any way identifies the branch instructions in a program as claimed.

The Office Action also mistakenly alleges that Coutant's col. 8, ll. 25-26 teach the limitations of replacing the target addresses of the branch instructions with the first address. This section of Coutant is wrongly applied because the entire paragraph makes clear that *load* instructions are examined, not *branch* instructions (col. 8, ll. 8-28). Coutant replaces a linkage table *load* instruction with a *noop*, and an *indirect load* instruction is replaced with a *direct load* instruction. Not only does this teaching fail to suggest *branch* instructions, but the teaching says nothing of the claimed replacing of a target address. Therefore, the Office Action fails to show that Coutant anticipates claim 1.

The rejection of claim 2 fails for the same general reasons as set forth above in the response to the rejection of claim 1. That is, the limitations of claim 2 involve the target address of a branch instruction, not linkage tables and load instructions.

Claim 15 is an apparatus claim, and to the extent that the limitations of claim 15 are similar to those of claim 1, the Office Action does not establish that Coutant anticipates claim 15 for at least the reasons set forth above.

Withdrawal of the §102 rejection of claims 1, 2, and 15 is respectfully requested since Coutant is not shown to anticipate these claims.

The Office Action uses Coutant as a basis in rejecting claims 3-14 under 35 U.S.C. §103(a). However, it is respectfully submitted that under 35 U.S.C. §103(c), Coutant should be removed as a reference because at the time of the present invention (with an application filing date of July 9, 2001), the present invention and Coutant were both owned, or subject to assignment to Hewlett-Packard Company. Therefore, Coutant is not eligible as prior art under §103, and the rejections should be withdrawn.

Even though the Coutant-based rejections under §103 should be withdrawn based on §103(c), further arguments are set forth below to make clear that the Office Action does not establish *prima facie* obviousness even if Coutant were not disqualified under §103(c).

The Office Action fails to establish that claims 3 - 5 are unpatentable under 35 U.S.C. §103(a) over Coutant, in view of admitted prior art. The rejection is respectfully traversed because the Office Action fails to show that all the limitations are suggested by the references, fails to provide a proper motivation for modifying the teachings of Coutant, and fails to show that the modification could be made with a reasonable likelihood of success.

The Office Action does not establish that Coutant suggests the limitations of claim 3, including, searching a symbol table for an entry having a symbolic name that is a derivation of the first name and reading a linkage stub address associated with the symbolic name; and replacing target addresses of branch instructions having target addresses equal to the linkage stub address with an address at which the code that implements the function is stored. As explained above in the response to the rejection of claim 1, the cited sections of Coutant involve *load* instructions, not *branch* instructions. Thus, the Office Action does not establish that Coutant suggests the further limitations of claims 3-5 that further refine the limitations of claim 1 that are related to branch instructions.

The alleged motivation for modifying Coutant is improper because it simply states the purpose of the cited claim limitation. The alleged motivation provides no evidence that Coutant's approach would be improved if a linkage table included a derivative of a named function along with a linkage stub address. Furthermore, the alleged motivation does not provide any evidence to support modifying Coutant from optimizing shared data accesses to optimizing branches to functions through stubs. The alleged motivation is conclusory and based on hindsight, and therefore, improper.

The rejection of claims 3-5 over Coutant and admitted prior art should be withdrawn because the Office Action fails to show all the limitations are suggested by the combination, fails to provide a proper motivation for combining the references, and fails to show that the combination could be made with a reasonable likelihood of success.

The Office Action fails to establish that claims 6, 7 and 11 are unpatentable under 35 U.S.C. 103(a) over Coutant in view of US Publication No. 2002/0026633 to Koizumi et al., (hereinafter "Koizumi"). The rejection is respectfully traversed because the Office Action fails to show that all the limitations are suggested by the references, fails to provide a proper motivation for modifying the teachings of Coutant with teachings of Koizumi, and fails to show that the combination could be made with a reasonable likelihood of success.

Claim 6 includes further limitations of replacing function entry points in the executable program with breakpoints, whereby breakpointed functions are generated; and upon encountering a breakpoint of a breakpointed function during program execution, identifying within the breakpointed function branch instructions that target linkage stub functions. The Office Action is in error in relying on various sections of Koizumi as suggesting these limitations.

For example, the Office Action cites Koizumi's page 17, para. 387 as suggesting these limitations. However, this section describes setting a flag to indicate a breakpoint at a program statement. This is not suggestive of *replacing* the function entry points as claimed. Furthermore, the cited para. 389 has no apparent relevance to the limitations of identifying instructions in a breakpointed function that target linkage stub functions. Para. 389 describes processing of an execute command. The described processing involves extracting the address of an instruction, for example, from a program counter, and checking whether a breakpoint is set at the determined address. This is clearly not suggestive of identifying instructions within a breakpointed function that target linkage stub functions (also note that neither Coutant nor Koizumi reference "instructions that target linkage stub functions"). Thus, the limitations of claims 6, 7, and 11 are not shown to be suggested by the Coutant-Koizumi combination

The alleged motivation for modifying Coutant with the cited teachings of Koizumi is improper because it is conclusory and based on hindsight. The alleged motivation simply restates Koizumi's function of targeting code for a particular machine, and no evidence is provided to indicate that Coutant's system would or even could benefit from this feature. Coutant's system only discusses optimizing remote data access. Supplanting Koizumi's setting of a breakpoint flag and determining of breakpoints by way of getting the current instruction address do not appear to have any particular relevance to Coutant's modifying of load instructions for optimizing remote data access. Furthermore, no evidence is provided to suggest that Coutant's system is in some way deficient and in need of improvement.


The rejection of claims 6-7 and 11 over the Coutant-Koizumi combination should be withdrawn because the Office Action fails to show all the limitations are suggested by the combination, fails to provide a proper motivation for combining the references, and fails to show that the combination could be made with a reasonable likelihood of success.

The rejection of claims 8 - 10 and 12 - 14 under 35 U.S.C. 103(a) as being unpatentable over the Coutant-Koizumi combination in view of admitted prior art is improper for at least the reasons set forth above. *Prima facie* obviousness is not established because all the limitations are not shown by the Office Action to be suggested by the combination, the alleged motivation is improper, and no reasonable likelihood of successfully making the combination has been shown.

No extension of time is believed to be necessary for consideration of this response. However, if an extension of time is required, please consider this a petition for a sufficient number of months for consideration of this response. If there are any additional fees in connection with this response, please charge Deposit Account No. 50-0996 (HPCO.049PA).

Respectfully submitted,

CRAWFORD MAUNU PLLC
1270 Northland Drive, Suite 390
Saint Paul, MN 55120
(651) 686-6633

By: 
Name: LeRoy D. Maunu
Reg. No.: 35,274